

Association_exercise

Damian R. Plichta

Introduction

Welcome to the exercise that accompanies lecture **Microbiome association analysis**. Throughout the exercises we will use different statistical tests to give you an idea of the basic tools and commands for data analysis in R that might come useful during your project work.

You will be working with microbiome data from Nielsen et al, 2014 that was published in Nature Biotechnology by our group (10.1038/nbt.2939). The dataset consists of:

- metadata - describes study participants
- speciec_matrix - summarizes microbial abundance in every study participant; it includes MGS'es (species; at least 700 genes) and CAG's (small genetic units; less than 700 genes)
- mgs_cag_geneCount - vector that summarizes number of genes in MGS'es and CAG's

The study included Spanish people - some with inflammatory bowel disease (IBD) - Crohn's disease (CD) or ulcerative colitis (UC), and Danish people - some with obesity. The paper itself did not focus on the human phenotype but you will analyze it also in that context.

You will notice that for plotting I am often using ggplot() command. I encourage you to make yourself familiar with it if you ever plan to do a lot of plotting in R. To give you a head start I made a small primer exercise where I explain ggplot syntax.

The exercises should be run on padawan using **R-3.1.2** The data can be accessed from:
/home/local1/27636/Ex_association/association_metagenomics.RData

ggplot primer

In ggplot a data frame is essential. We start by creating example_df that contains columns (variables) describing height, weight and sex information for four individuals. With that we are ready to use ggplot.

In the ggplot() command you start by specifying the data frame with input data and aes(). In aes() you define which column from the data frame should be used as x and y variable. Additionally, in aes() you can specify which column should be used for coloring points, lines or bars in the plots you create.

Now, to plot a scatterplot apart from defining ggplot() you need to add '+ geom_point()'. To make a barplot you need to add '+ geom_bar()'. Try following examples:

```
library(ggplot2) # plotting package
example_df <- data.frame( Name=c("S1", "S2", "S3", "S4"), Height=c(183, 178, 159, 192), Weight=c(78, 65, 70, 84), Sex=c("M", "M", "F", "F"))

example_df # print the table to the terminal
```

```
##      Name Height Weight Sex
## 1     S1     183     78   M
## 2     S2     178     65   M
## 3     S3     159     70   F
## 4     S4     192     84   F
```

```
# plots a scatterplot:
ggplot(example_df, aes(x=Weight,y=Height,color=Sex)) + geom_point()
# plots a barplot:
ggplot(example_df, aes(x=Name,y=Height,fill=Sex)) + geom_bar(stat="identity")
```

Play with the line producing a barplot:

- change 'fill' to 'color' - what is the difference?
- add '+ facet_grid(. ~ Sex, scales="free_x")' at the end of the line - what is the difference?

You can read more about the basic ggplot plotting here: <http://www.cookbook-r.com/Graphs/index.html> (<http://www.cookbook-r.com/Graphs/index.html>)

Data - loading and inspection

```
library(ggplot2) # plotting package
library(plyr) # data analysis package
library(reshape2) # data re-shaping package
load("association_metagenomics.RData") # input data

# check what kind of variables were loaded
ls()
```

```
## [1] "example_df"          "metadata"             "mgs_cag_geneCount"
## [4] "speciec_matrix"
```

Using ls() command you listed all objects present in your R environment. See how they look like by printing them to screen or running str() / lenght() command. Inspect metadata, speciec_matrix and mgs_cag_geneCount.

Q1: How many individuals are included in the dataset? How many MGS'es/CAG's altogether?

Metadata - categorical variables

The metadata consists of 393 rows, each row describing nationality, gender, age, BMI, health status and MGSrichness (number of gut microorganisms) of a study participant. The column Health_Status has been made with IBD patients in mind, that's why it only contains information about Spanish individuals. If you print it together with the Nationality column, you will see that the Danish individuals have NAs.

```
head(metadata[,c("Nationality","Health_Status")]) # print the top of the data frame
```

```
##           Nationality Health_Status
## MH0001          DK          <NA>
## MH0002          DK          <NA>
## MH0003          DK          <NA>
## MH0004          DK          <NA>
## MH0005          DK          <NA>
## MH0006          DK          <NA>
```

```
tail(metadata[,c("Nationality","Health_Status")]) # print the bottom of the data frame
```

```
##           Nationality Health_Status
## V1.UC6.0          ES    HEALTHYREL
## V1.UC61.0         ES    HEALTHYREL
## V1.UC62.0         ES    HEALTHYREL
## V1.UC7.0          ES    HEALTHYREL
## V1.UC8.0          ES    HEALTHYREL
## V1.UC9.0          ES    HEALTHYREL
```

Let's change the NAs in Danish individuals Healthy. We will do it by subsetting the elements in that column that are NAs with 'Healthy'. Notice that we identify NA by using command `is.na()` that returns TRUE/FALSE.

To see how `is.na()` works, try printing '`is.na(metadata$Health_Status)`' before and after executing following commands:

```
metadata$Health_Status[ is.na(metadata$Health_Status) ] <- "Healthy" # print the top of the data frame
table(metadata$Health_Status)
```

```
##
##           CD      Healthy      HEALTHY HEALTHYREL          UC
##           21         177          24          47         124
```

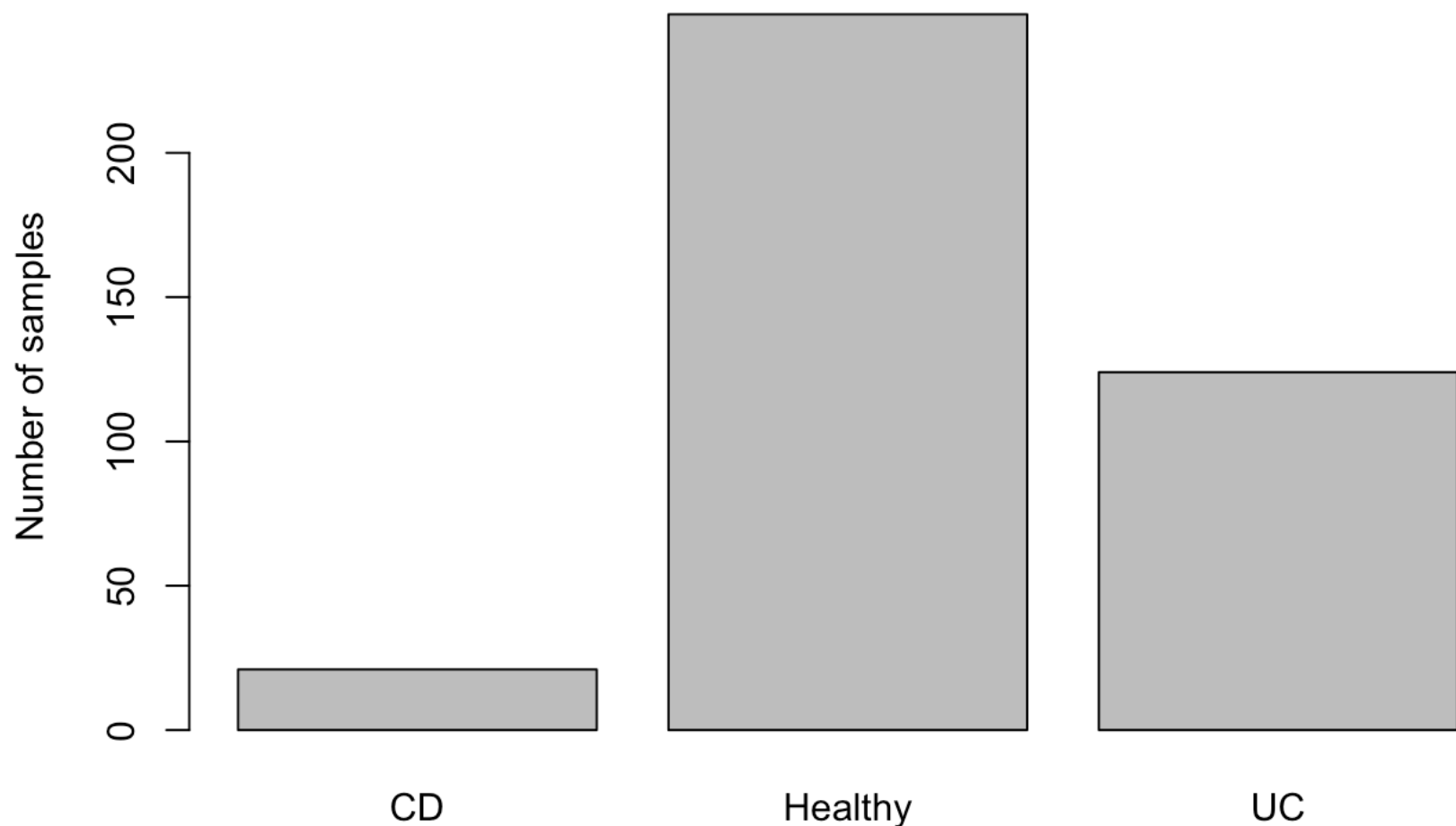
Command `table()` is useful when counting occurrence of each element in a vector. You noticed that column `Health_Status` has some additional categories representing healthy individuals, i.e. 'HEALTHY' and 'HEALTHYREL'. We need to remove this inconsistency by changing them to 'Healthy'.

```
metadata$Health_Status[ grep( "H", metadata$Health_Status ) ] <- "Healthy"
```

Q2: In the commands above we used function `grep`. Explain what it does - you can inspire yourself by looking at R help (`?grep`).

Q3: How many UC, CD, healthy individuals are in the dataset? Is this a balanced design?

```
barplot(table(metadata$Health_Status), ylab="Number of samples")
```



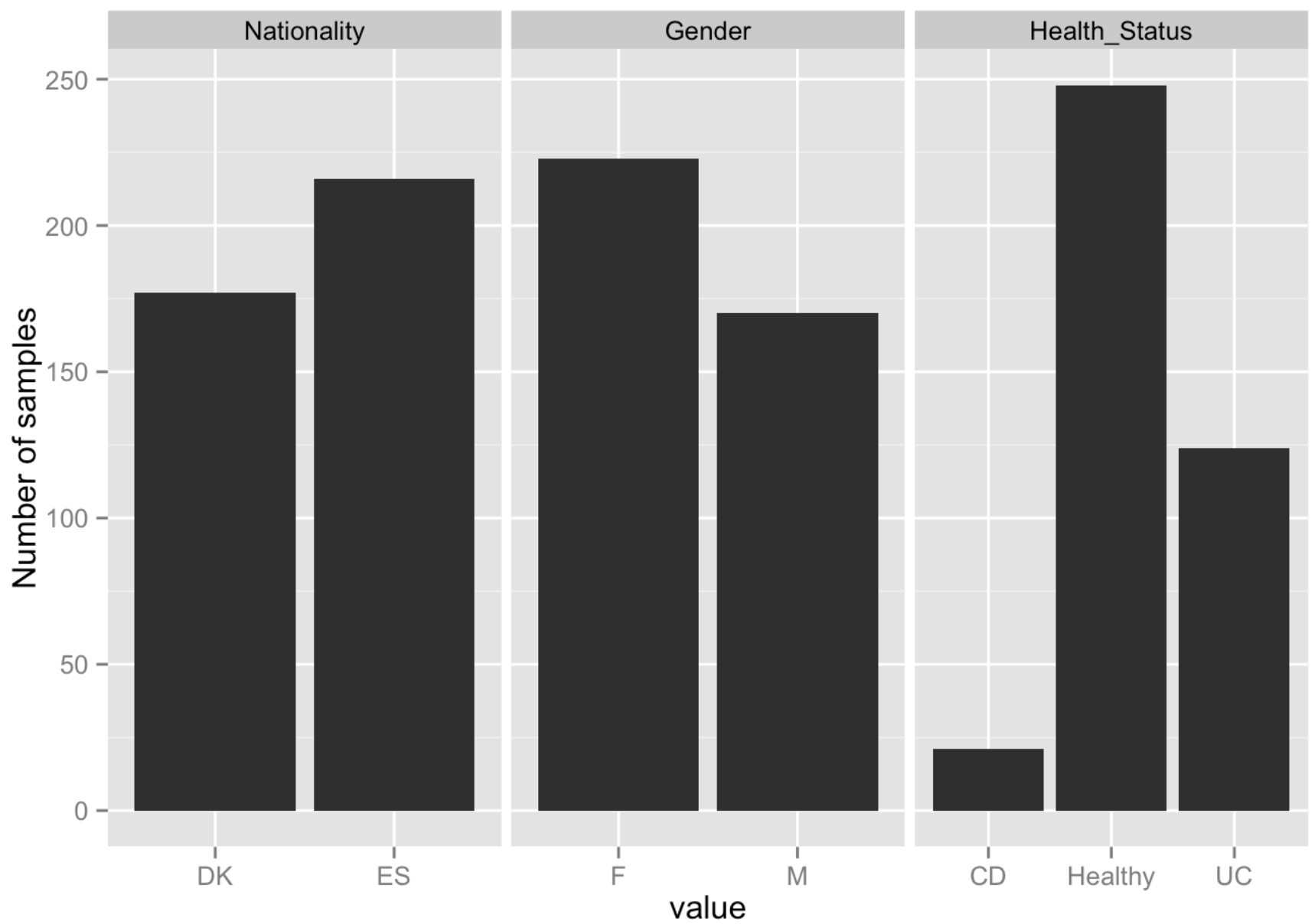
Q4: How would you plot barplots with sample counts for column Gender and Nationality.

Q5: How many Danes and Spaniards took part in the study? How many males and females?

Let's take a look at the plotting alternative with ggplot. To that we need to modify the data frame a bit. We will use `melt()` function from package `reshape2` and `ddply` from package `plyr`. In the code I included links to the websites where the mechanism of `melt` and `ddply` is explained.

```
sub_metadata <- metadata[,c("BaseID", "Nationality", "Gender", "Health_Status")]
sub_metadata <- melt(sub_metadata, id.vars="BaseID") # transforms the data from wide to long format; see http://seananderson.ca/2013/10/19/reshape.html
sub_metadata_summary <- ddply(sub_metadata, c("variable", "value"), summarise, count_elements=length(value)) # summarizes data frame by calculating specified values; see http://seananderson.ca/2013/12/01/plyr.html

ggplot(sub_metadata_summary, aes(x=value, y=count_elements)) + geom_bar(stat="identity") + ylab("Number of samples") + facet_grid( . ~ variable, scales="free_x" )
```

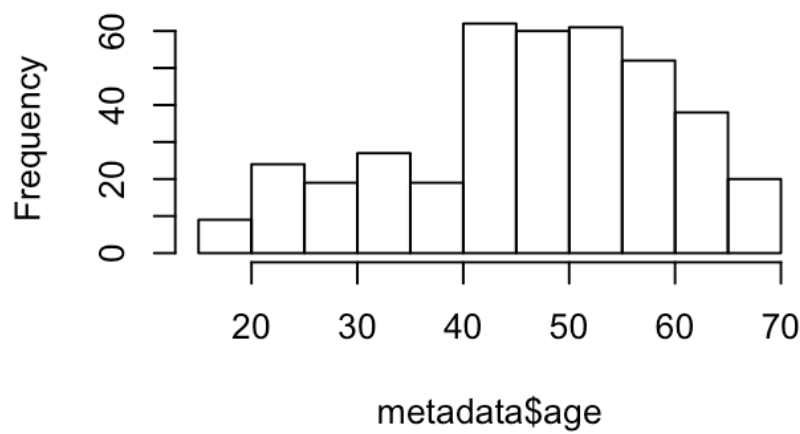


Metadata - continuous variables

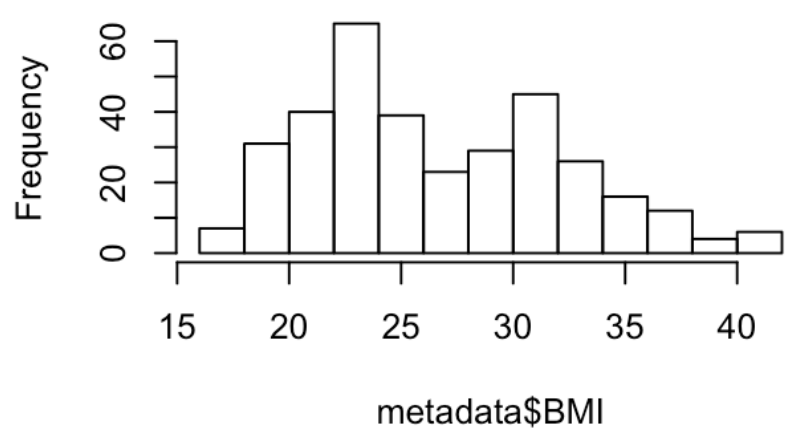
Let's have a look at the continuous variables - age, BMI and MGSrichness. By plotting their histograms we can investigate their distributions.

```
par(mfrow=c(2,2))  
hist(metadata$age)  
hist(metadata$BMI)  
hist(metadata$MGSrichness)
```

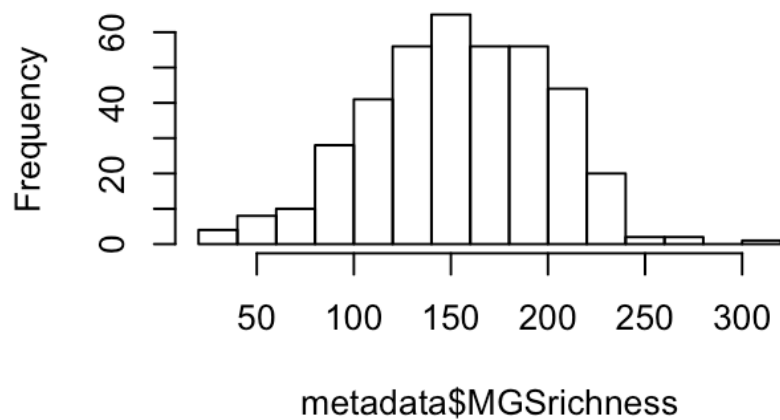
Histogram of metadata\$age



Histogram of metadata\$BMI



Histogram of metadata\$MGSRichness



As you remember, we are dealing with individuals from two countries - Denmark and Spain. Re-plot the histograms including information about the 'Nationality'. You can do that in ggplot. In `aes()` you need only to specify 'x' and 'fill': `aes(x=age, fill=Nationality)`. To plot histogram you need to use `'+ geom_histogram(position="identity", alpha=0.5)'`.

Q6: Inspect distribution of the continuous variables, colored/plotted according to nationality. Which one looks Gaussian/normal like?

Q7: In you opinion, is the BMI distribution among Danes representative for the population? What can be a purpose of a design like that?

Microbiome data

The microbiome data in `speciec_matrix` is a matrix containing abundance information for each MGS (species) and CAG (small genetic unit) across all samples. As you remember from earlier lectures, MGS'es contain at least 700 genes. Can you count how many MGS'es and CAG's are in the dataset? You can do that either by using R object `mgs_cag_geneCount` or by checking the rownames in `speciec_matrix` using function `grep()`.

Q8: How many MGS'es did you count? How many CAG's?

Let's plot the size distribution of MGS'es and CAG's. We will do it again using ggplot and for that we first need to create a data frame.

```
mgs_cag_geneCount_df <- data.frame( geneCount=mgs_cag_geneCount, ID=names(mgs_cag_geneCount), stringsAsFactors=F)
mgs_cag_geneCount_df$Species <- ifelse( grepl("M", mgs_cag_geneCount_df$ID), "Yes" , "No" )
head(mgs_cag_geneCount_df)
```

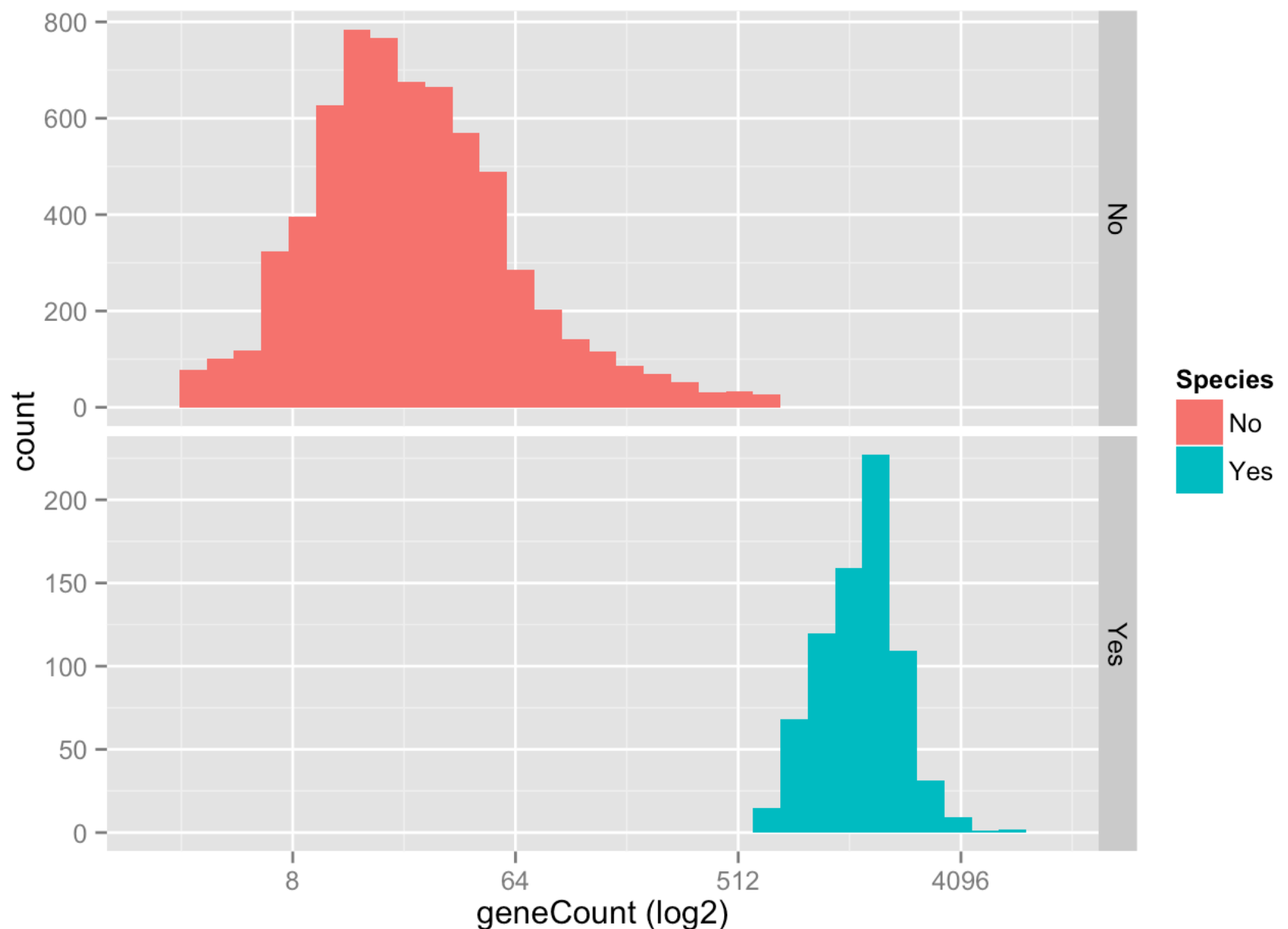
```
##           geneCount      ID Species
## MGS:1          2915 MGS:1      Yes
## MGS:2          2608 MGS:2      Yes
## MGS:3          2450 MGS:3      Yes
## MGS:4          3523 MGS:4      Yes
## MGS:5          2981 MGS:5      Yes
## MGS:6          2271 MGS:6      Yes
```

```
tail(mgs_cag_geneCount_df)
```

```
##           geneCount      ID Species
## CAG:7376           5 CAG:7376      No
## CAG:7377           7 CAG:7377      No
## CAG:7378           9 CAG:7378      No
## CAG:7379           5 CAG:7379      No
## CAG:7380           5 CAG:7380      No
## CAG:7381           4 CAG:7381      No
```

```
ggplot(mgs_cag_geneCount_df, aes(x=geneCount, fill=Species)) + geom_histogram() +
facet_grid( Species ~ . , scales="free_y" ) + scale_x_continuous(trans = "log2") +
xlab("geneCount (log2)")
```

```
## stat_bin: binwidth defaulted to range/30. Use 'binwidth = x' to adjust this.
## stat_bin: binwidth defaulted to range/30. Use 'binwidth = x' to adjust this.
```



In the commands above we used function `grepl()` that is similar to `grep()`, but produces TRUE/FALSE as well as function `ifelse()`. Check in R help how `ifelse` works by typing `?ifelse`.

Microbiome association analysis

Association A - species richness vs age

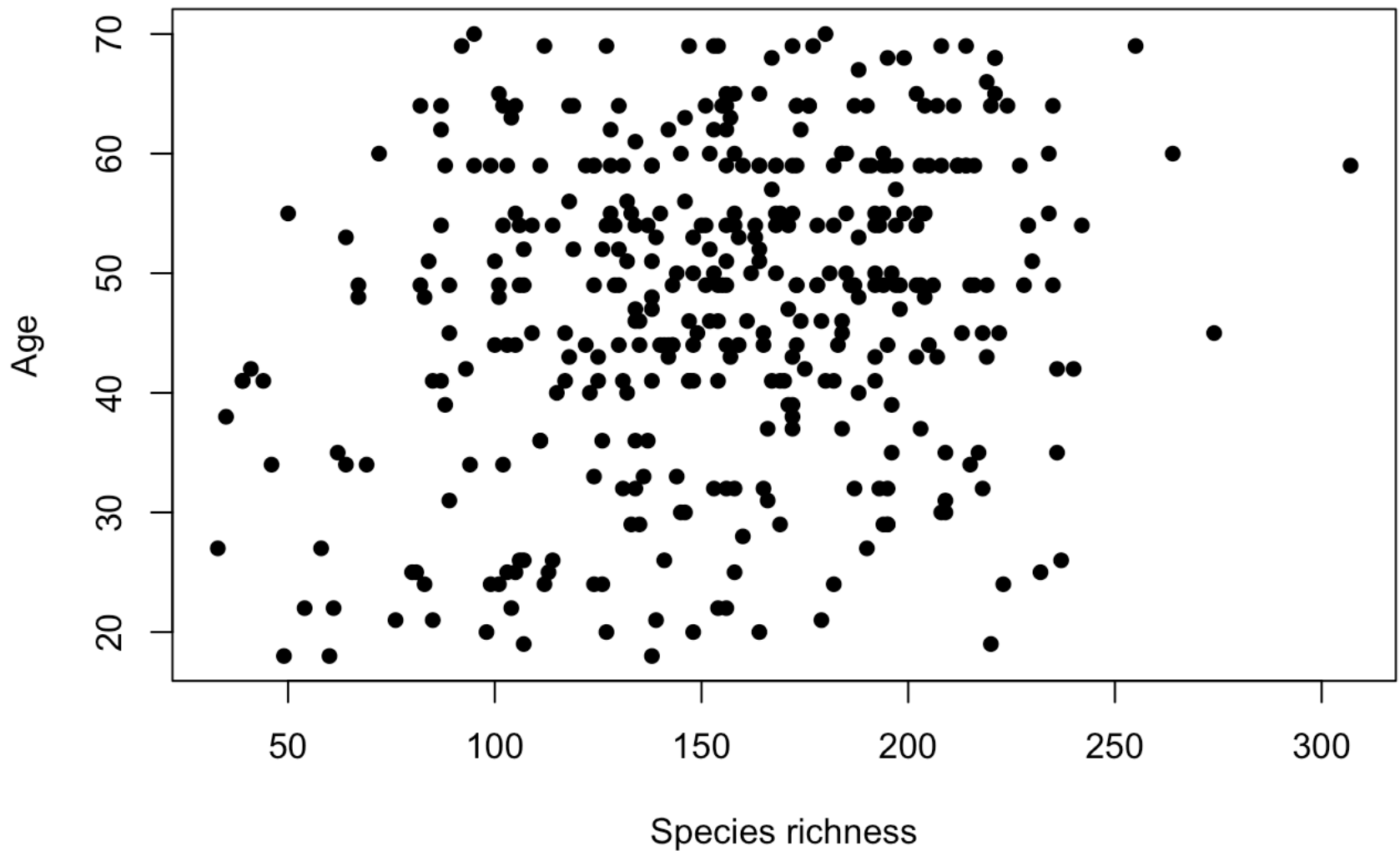
Species richness is estimated as the number of MGS'es detected in an individual. The object `metadata` contains column 'MGSrichness' with that estimate. How would you calculate it if you didn't have that information? You can use `speciec_matrix` for that purpose.

```
species_richness <- colSums(speciec_matrix[mgs_cag_geneCount>=700,] > 0)
```

Q9: Explain what happens in that command? Does the output agrees with the metadata column 'MGSrichness'?

Let's look at the association between age and richness. Usually you would try to visualize data to see if there is any signal. In this case both variables are continuous so you can use a scatterplot for the purpose.

```
plot(metadata$MGSrichness, metadata$age, pch=16, xlab="Species richness", ylab="Age")
```

Q10: Comment on the richness to age relationship.

Let's calculate a correlation coefficient for relationship between richness and age. This is done using `cor()` command.

```
cor(metadata$MGSrichness, metadata$age) # returns correlation coefficient
```

```
## [1] NA
```

```
cor(metadata$MGSrichness, metadata$age, use="complete.obs") # returns correlation coefficient
```

```
## [1] 0.2362206
```

```
cor.test(metadata$MGSrichness, metadata$age, use="complete.obs") # returns correlation coefficient and pvalue
```

```
##
## Pearson's product-moment correlation
##
## data: metadata$MGSrichness and metadata$age
## t = 4.7947, df = 389, p-value = 2.324e-06
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.1403332 0.3277181
## sample estimates:
##          cor
## 0.2362206
```

Q11: Why did the first `cor()` command produce NA but not the second one? Look at R help page for `cor()` (?cor). Can you identify what and where went wrong?

Q12: How strong is the association between richness and age? Report correlation coefficient and pvalue. Can you comment it in relation to paper [dx.doi.org/10.1038/nature11053](https://doi.org/10.1038/nature11053) by Yatsunenko et al, 2012 (especially Figure 2)?

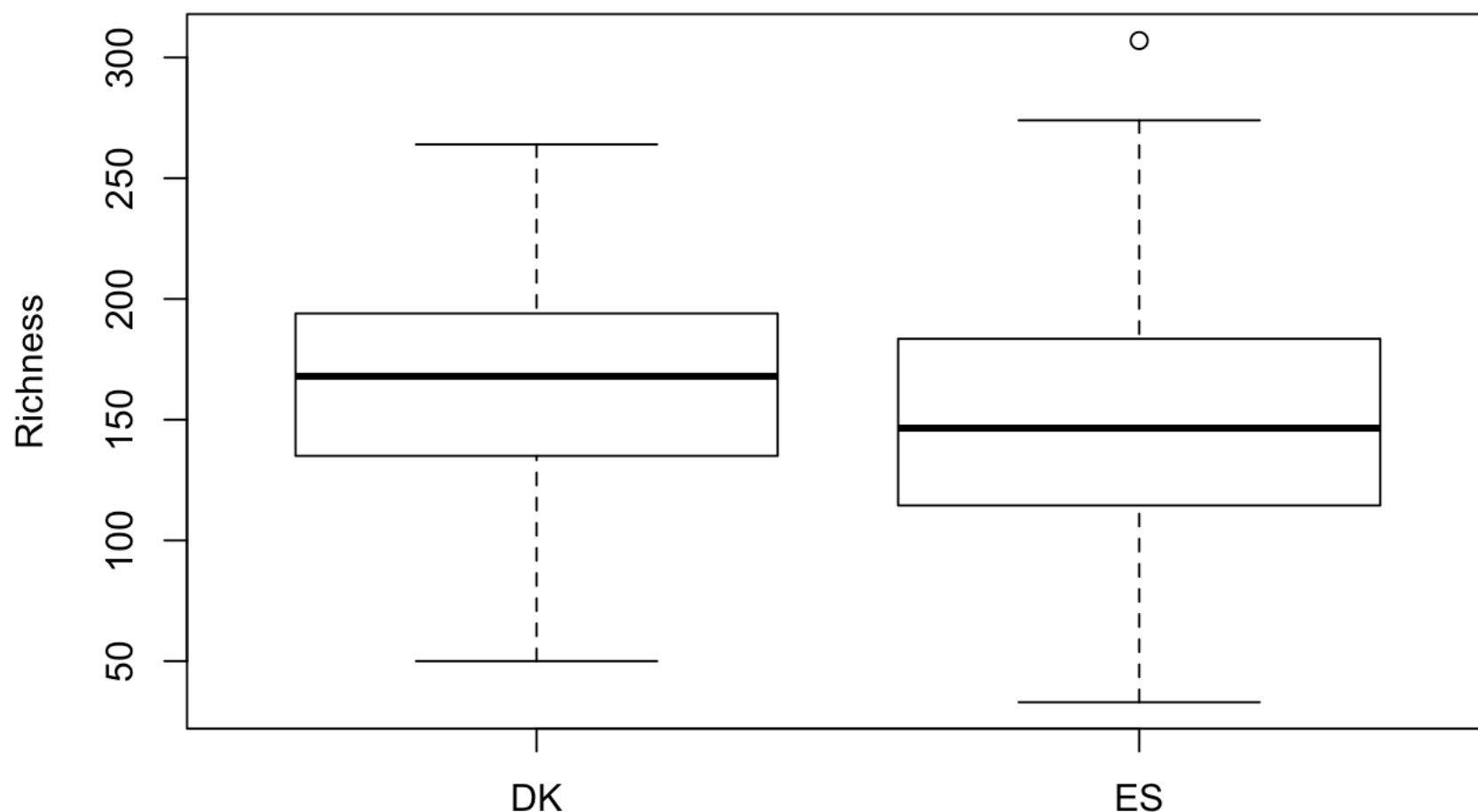
Try plotting the same relationship in ggplot. You can visualize the age to richness trend by adding a regression line to a plot:

```
ggplot(metadata, aes(x=MGSrichness, y=age)) + geom_point() + xlab("Species richness") + ylab("Age") + geom_smooth(method=lm, se=FALSE)
```

Association B - species richness vs nationality

We will now look at the relationship between richness and nationality. Nationality is a discrete variable, that's why for visualization we will use a boxplot.

```
boxplot( metadata$MGSrichness ~ metadata$Nationality, ylab="Richness")
```



The plot suggests that there is a difference between the two groups in species richness. We can test it using t-test or one-way anova. Let's try the latter. One-way anova tests if there is a significant difference between the means of two or more groups. It is usually followed by a post-hoc test to identify exactly which of the two or more groups is significantly different. We will compute anova model using command `aov()`. As a post-hoc test we will use Tukey's procedure that is implemented in command `TukeyHSD()`.

Q13: Why would it make sense to use t-test instead of one-way anova in this case?

```
richness_gender_nationality <- aov( metadata$MGSrichness ~ metadata$Nationality )
summary(richness_gender_nationality)
```

```
##              Df Sum Sq Mean Sq F value    Pr(>F)
## metadata$Nationality    1   25235     25235    11.73 0.000681 ***
## Residuals              391  841263      2152
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
TukeyHSD(richness_gender_nationality)
```

```
## Tukey multiple comparisons of means
## 95% family-wise confidence level
##
## Fit: aov(formula = metadata$MGSrichness ~ metadata$Nationality)
##
## $`metadata$Nationality`
##          diff          lwr          upr          p adj
## ES-DK -16.10578 -25.3518 -6.859755 0.0006807
```

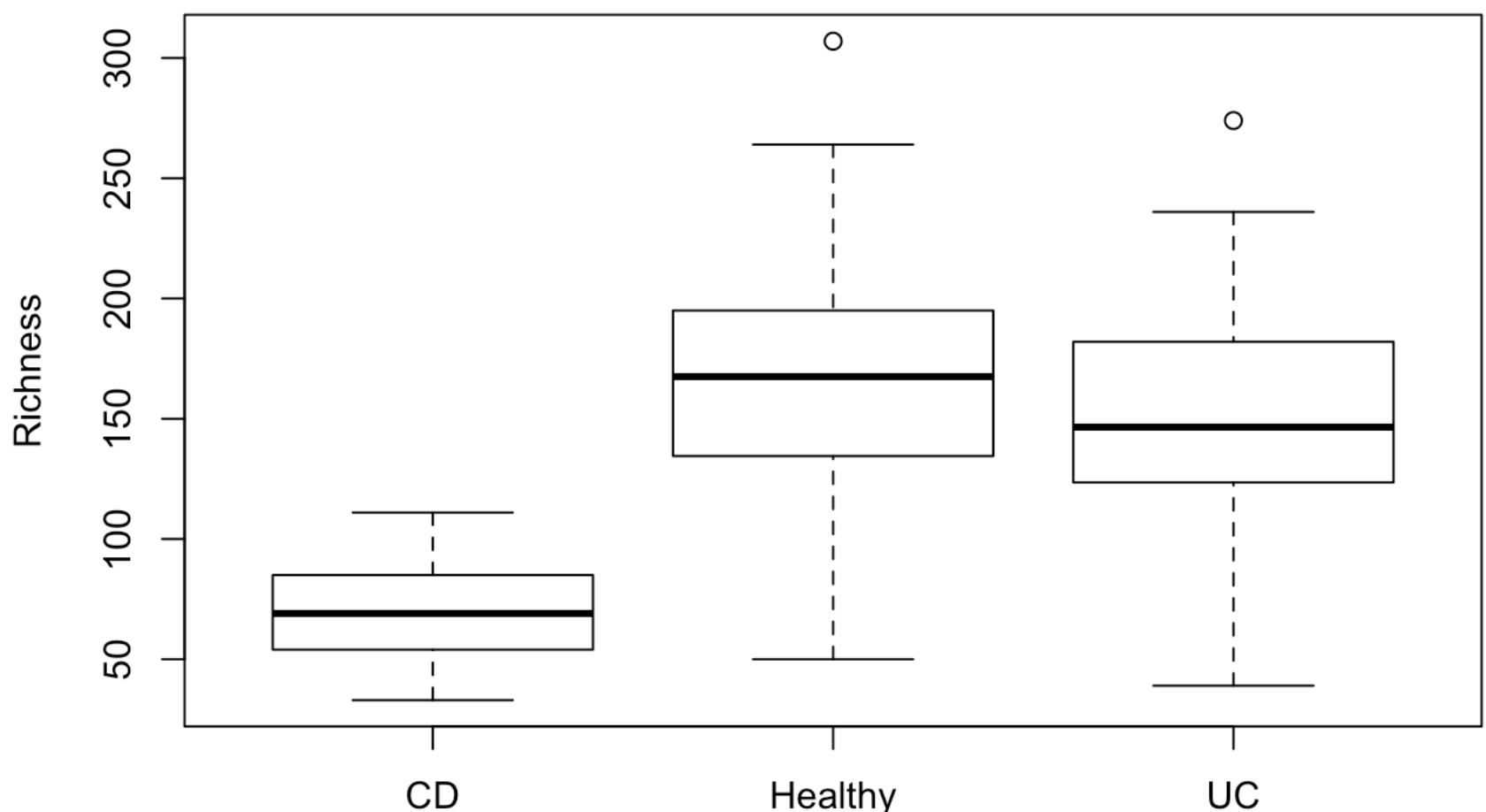
You can run t-test using `t.test()` command. Compare the result with one-way anova.

Q14: Is there a significant difference between Danes and Spaniards? Do you trust it? Read the abstract and look at the graphical abstract in Gevers et al, 2014 ([http://www.cell.com/cell-host-microbe/abstract/S1931-3128\(14\)00063-8](http://www.cell.com/cell-host-microbe/abstract/S1931-3128(14)00063-8) ([http://www.cell.com/cell-host-microbe/abstract/S1931-3128\(14\)00063-8](http://www.cell.com/cell-host-microbe/abstract/S1931-3128(14)00063-8))).

Association C - species richness vs disease

As the inflammatory bowel disease, especially Crohn's disease (CD), is associated with microbial dysbiosis we will look at the association of that phenotype and richness. First the visual inspection of the data:

```
boxplot( metadata$MGSrichness ~ metadata$Health_Status, ylab="Richness")
```



Q15: Considering this plot and previous association between the richness and nationality, do you think that Spanish people on average have a lower species richness? How could you test it?

Let's make the statistical test to see if the difference between CD, UC and Healthy is significant. Again, we will use one-way anova and a post-hoc test using Tukey's procedure.

```
richness_disease_nationality <- aov( metadata$MGSrichness ~ metadata$Health_Status
)
summary(richness_disease_nationality)
```

```
##              Df Sum Sq Mean Sq F value Pr(>F)
## metadata$Health_Status      2 172040    86020    48.31 <2e-16 ***
## Residuals                   390 694458    1781
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
TukeyHSD(richness_disease_nationality)
```

```
##      Tukey multiple comparisons of means
##      95% family-wise confidence level
##
## Fit: aov(formula = metadata$MGSrichness ~ metadata$Health_Status)
##
## $`metadata$Health_Status`
##              diff              lwr              upr              p adj
## Healthy-CD    92.97158    70.40857 115.534598 0.0000000
## UC-CD         78.27803    54.85085 101.705213 0.0000000
## UC-Healthy   -14.69355   -25.61276  -3.774337 0.0047388
```

Q16: Comment on the results. How significant is the difference between CD and Healthy or UC and Healthy?

Association D - Crohn's disease vs MGS abundnace

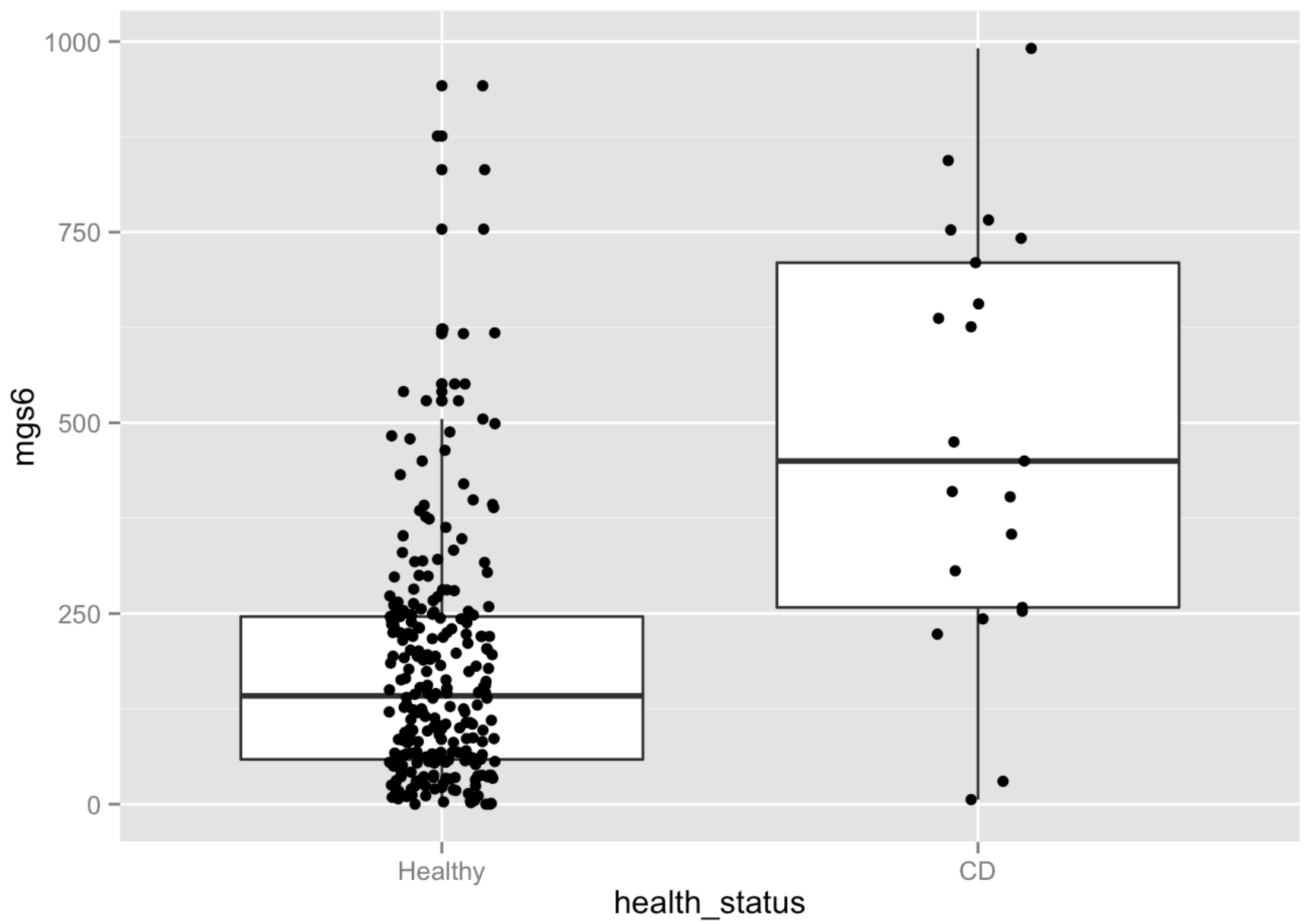
As we learned that Crohn's disease is associated with changes in the gut microbiome, let's see which microbial species (MGS) changes their abundance compared to the healthy individuals.

We will only look at the CD and Healthy part of the study, so we need to create a corresponding index vector. Let's visualize MGS:6 (*Bacteroides vulgatus*) - it's abundance in Healthy and CD patients as well it's relation to richness.

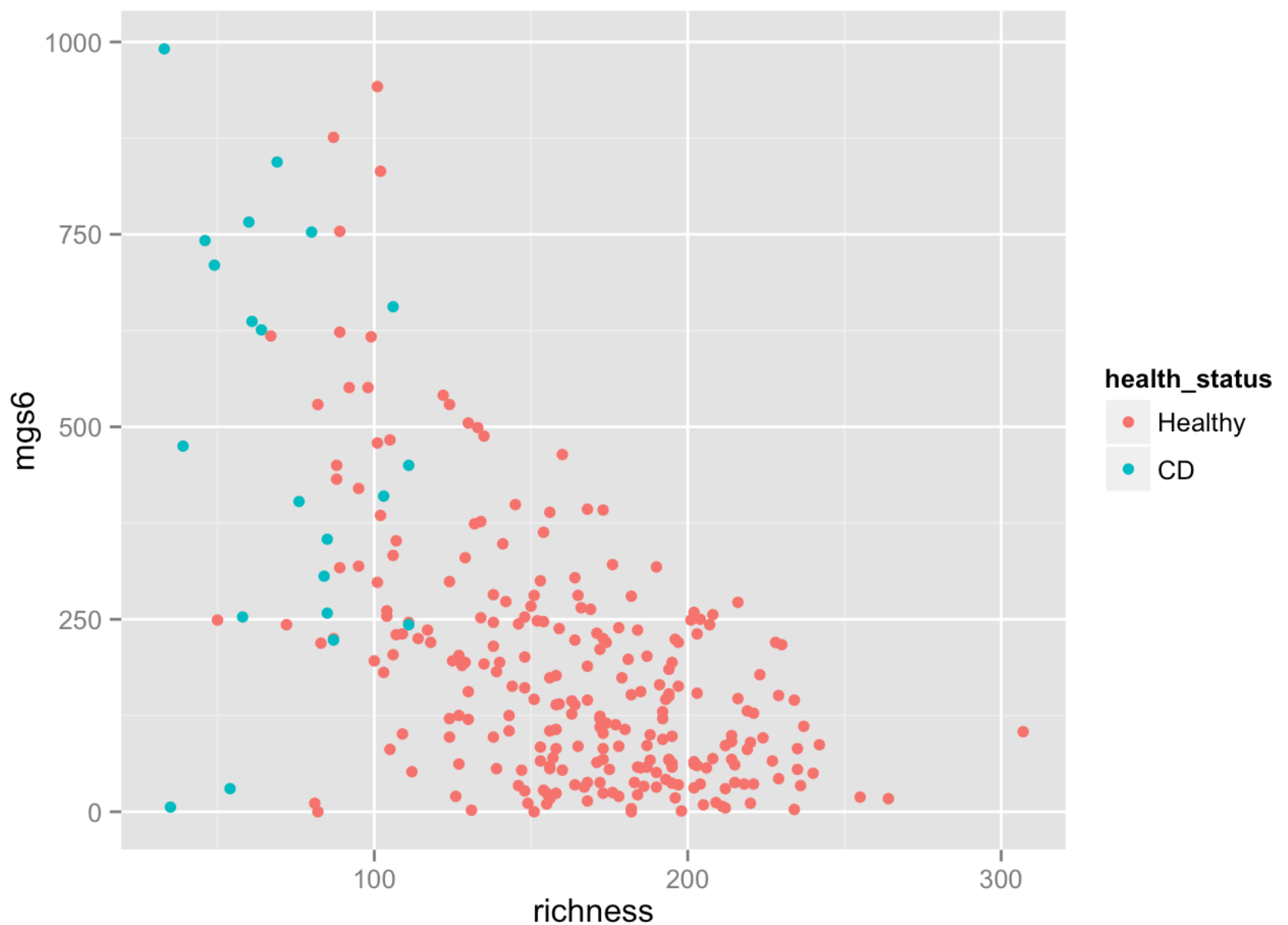
```
i_CD_Healthy <- metadata$Health_Status %in% "CD" | metadata$Health_Status %in% "Healthy"
metadata$Health_Status <- relevel(factor(metadata$Health_Status), "Healthy")

mgs6_df <- data.frame( mgs6=speciec_matrix["MGS:6",i_CD_Healthy], health_status=metadata$Health_Status[i_CD_Healthy], richness=metadata$MGSrichness[i_CD_Healthy] )

ggplot(mgs6_df, aes(x=health_status,y=mgs6)) + geom_boxplot() + geom_jitter(position=position_jitter(width=.1, height=0))
```



```
ggplot(mgs6_df, aes(x=richness,y=mgs6,color=health_status)) + geom_point()
```



Q17: Comment on the plots - consider MGS:6 abundance to richness relationship.

Abundance of MGS:6 is to some extent correlated to richness; we know that richness is also associated with the Crohn's disease. When testing species abundance to disease status in the linear model we will add a term for richness. The linear model is called using command `lm()`. By including richness term in the model we can account for its effect and get a more reliable estimate of relationship between Crohn's disease and species abundance.

```
summary(lm(speciec_matrix["MGS:6",i_CD_Healthy] ~ metadata$Health_Status[i_CD_Healthy] ))
```

```
##
## Call:
## lm(formula = speciec_matrix["MGS:6", i_CD_Healthy] ~ metadata$Health_Status[i_C
D_Healthy])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -476.67 -120.99  -34.99   70.01  763.01
##
## Coefficients:
##                                Estimate Std. Error t value
## (Intercept)                   178.99      10.97   16.319
## metadata$Health_Status[i_CD_Healthy]CD  303.68      39.26    7.736
##                                Pr(>|t|)
## (Intercept)                   < 2e-16 ***
## metadata$Health_Status[i_CD_Healthy]CD 2.11e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 172.7 on 267 degrees of freedom
## Multiple R-squared:  0.1831, Adjusted R-squared:  0.18
## F-statistic: 59.84 on 1 and 267 DF,  p-value: 2.114e-13
```

```
summary(lm(speciec_matrix["MGS:6",i_CD_Healthy] ~ metadata$Health_Status[i_CD_Heal
thy] + metadata$MGSrichness[i_CD_Healthy] )) # richness term included
```

```
##
## Call:
## lm(formula = speciec_matrix["MGS:6", i_CD_Healthy] ~ metadata$Health_Status[i_C
D_Healthy] +
##      metadata$MGSrichness[i_CD_Healthy])
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -550.23  -92.43  -28.57   75.20  634.70
##
## Coefficients:
##                                Estimate Std. Error t value
## (Intercept)                   512.3367      37.6120   13.622
## metadata$Health_Status[i_CD_Healthy]CD 114.9447      39.9958    2.874
## metadata$MGSrichness[i_CD_Healthy]      -2.0300      0.2215   -9.165
##                                Pr(>|t|)
## (Intercept)                   < 2e-16 ***
## metadata$Health_Status[i_CD_Healthy]CD  0.00438 **
## metadata$MGSrichness[i_CD_Healthy]      < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 150.9 on 266 degrees of freedom
## Multiple R-squared:  0.3791, Adjusted R-squared:  0.3745
## F-statistic: 81.22 on 2 and 266 DF,  p-value: < 2.2e-16
```


Q18: Comment the difference in significance of CD status to abundance association without and with accounting for richness.

Let's run the linear model across all MGS'es. We will do it using a for-loop and we will save pvalues in a data frame. For reporting significantly associated MGS'es we will only use the pvalues calculated using linear model that accounts for richness. Finally, we will correct the pvalues for multiple testing using Benjamini & Hochberg method implemented in p.adjust().

```
mgs_names <- names(mgs_cag_geneCount[ mgs_cag_geneCount > 700 ]) # get species names
stat_out_lm_df <- data.frame() # data frame to store results
for( i in mgs_names){ # we are looping through MGS names

  stat_out_abundance <- summary(lm(speciec_matrix[i,i_CD_Healthy] ~ metadata$Health_Status[i_CD_Healthy] ))
  stat_out_abundance_richness <- summary(lm(speciec_matrix[i,i_CD_Healthy] ~ metadata$Health_Status[i_CD_Healthy] + metadata$MGSrichness[i_CD_Healthy] )) # richness term included

  # with stat_out_abundance$coefficients[2,4] we access the p.value estimate - that's the output we are interested in:
  stat_out_lm_df <- rbind(stat_out_lm_df, c(stat_out_abundance$coefficients[2,4] , stat_out_abundance_richness$coefficients[2,4]))

  # uncomment to monitor progress
  #cat(i, "\n") # prints a name of MGS - allows to monitor progress of the calculation
}
rownames(stat_out_lm_df) <- mgs_names
colnames(stat_out_lm_df) <- c("Pvalue_CD", "Pvalue_CD_Richness")
# remove MGS with missing estimates - occurs for MGS'es not detected in any sample
stat_out_lm_df <- stat_out_lm_df[!is.nan(stat_out_lm_df[,1]),]

# Order according to pvalue
stat_out_lm_df <- stat_out_lm_df[order(stat_out_lm_df$Pvalue_CD_Richness),]

# add FDR value - multiple testing correction
stat_out_lm_df$Pvalue_CD_Richness_BH <- p.adjust( stat_out_lm_df$Pvalue_CD_Richness, method="BH" )

sig_mgs_lm <- rownames(stat_out_lm_df)[stat_out_lm_df$Pvalue_CD_Richness_BH < 0.05]
```

Q19: How many MGS'es have their abundance significantly associated with Crohn's disease? What's the difference before and after correcting for multiple testing?

Bonus: association - richness vs disease

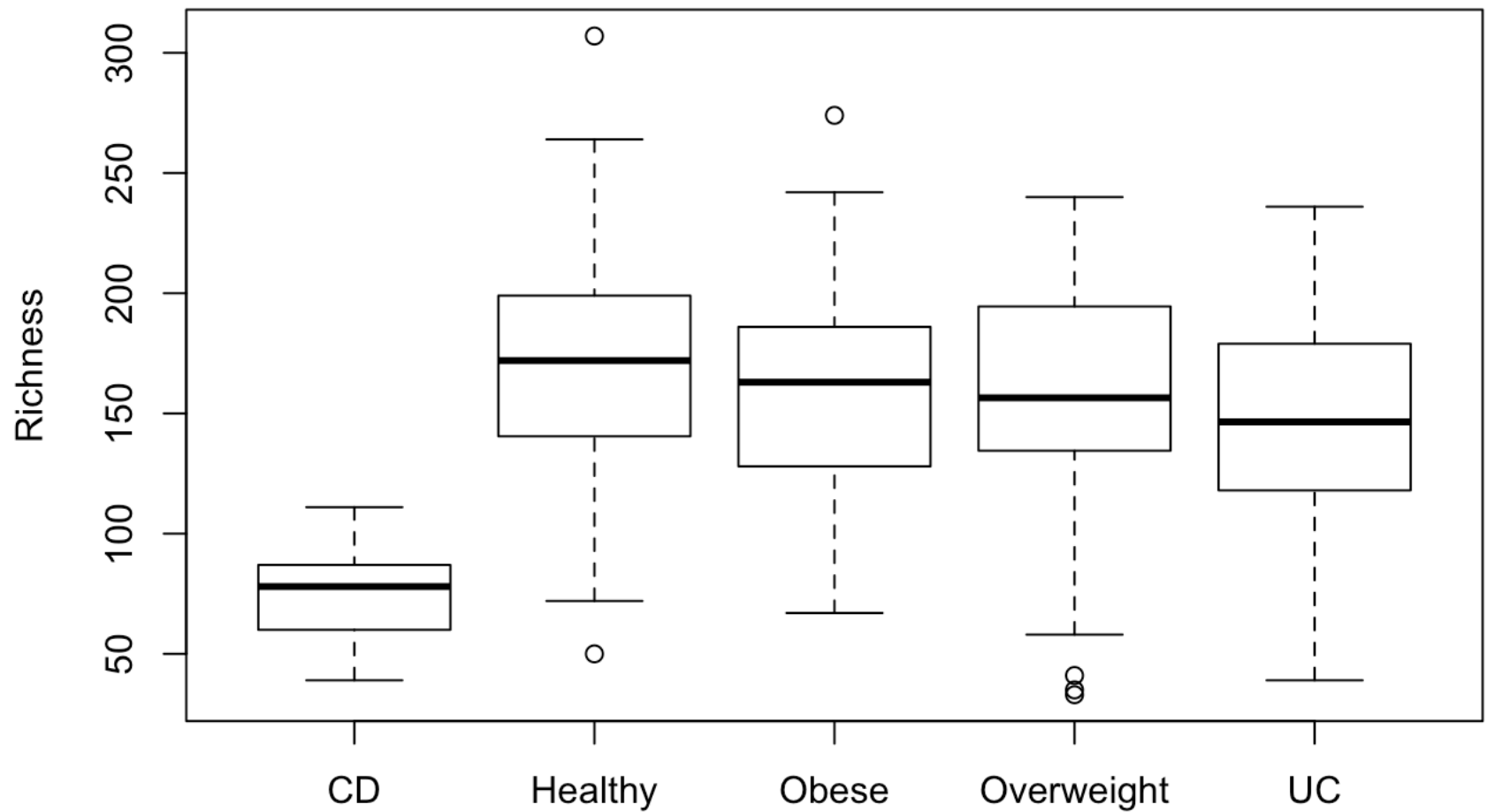
You might have been wondering why we didn't test richness association with overweight or obese phenotype. To do that we need to supplement 'Health_Status' variable with 'Overweight' and 'Obese' categories. We do that based on BMI scores; BMI 25-30 is classified as overweight and BMI>30 as obese. Notice that we create a new column in metadata called 'Health_Extended'.

```

metadata$Health_Extended <- as.character(metadata$Health_Status)
metadata$Health_Extended[metadata$BMI > 25] <- "Overweight"
metadata$Health_Extended[metadata$BMI > 30] <- "Obese"

boxplot( metadata$MGSrichness ~ metadata$Health_Extended, ylab="Richness")

```



```

richness_disease_extended <- aov( metadata$MGSrichness ~ metadata$Health_Extended
)
summary(richness_disease_extended)

```

```

##              Df Sum Sq Mean Sq F value    Pr(>F)
## metadata$Health_Extended    4 140202    35050    18.73 4.36e-14 ***
## Residuals              388  726296     1872
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

TukeyHSD(richness_disease_extended)

```

```
## Tukey multiple comparisons of means
## 95% family-wise confidence level
##
## Fit: aov(formula = metadata$MGSrichness ~ metadata$Health_Extended)
##
## $`metadata$Health_Extended`
##
```

| | diff | lwr | upr | p adj |
|--------------------|-------------|-----------|------------|-----------|
| Healthy-CD | 92.8600427 | 62.58941 | 123.130672 | 0.0000000 |
| Obese-CD | 81.6044852 | 51.43646 | 111.772506 | 0.0000000 |
| Overweight-CD | 82.5277778 | 51.28043 | 113.775129 | 0.0000000 |
| UC-CD | 70.6000000 | 39.98397 | 101.216027 | 0.0000000 |
| Obese-Healthy | -11.2555575 | -27.50934 | 4.998221 | 0.3201695 |
| Overweight-Healthy | -10.3322650 | -28.51118 | 7.846646 | 0.5255751 |
| UC-Healthy | -22.2600427 | -39.33098 | -5.189104 | 0.0036173 |
| Overweight-Obese | 0.9232926 | -17.08424 | 18.930827 | 0.9999120 |
| UC-Obese | -11.0044852 | -27.89281 | 5.883836 | 0.3833173 |
| UC-Overweight | -11.9277778 | -30.67619 | 6.820633 | 0.4084718 |

Q: Is the difference between Obese and Healthy significant? What about Overweight and Healthy?

Bonus: Association - Crohn's disease vs MGS abundance - non-parametric test

As mentioned during the lecture, you can also use non-parametric tests for estimating associations between species abundance and other parameters. Here is an example of the Crohn's disease to MGS abundance analysis using Wilcox test (Mann-Whitney). We will perform it only on species that are observed in at least 10 Crohn's disease individuals.

```
speciesCD <- rowSums(speciec_matrix[ mgs_cag_geneCount>700 ,metadata$Health_Status
== "CD"] > 0)
speciesCD <- names(speciesCD)[ speciesCD >= 10 ]

non_parametric_CD <- c()
for(i in speciesCD){
  test_out <- wilcox.test(speciec_matrix[i,i_CD_Healthy] ~ metadata$Health_Status[i_CD_Healthy])
  non_parametric_CD <- c(non_parametric_CD, test_out$p.value)
}

stat_out_wilcox_df <- data.frame( MGS=speciesCD, pvalue=non_parametric_CD, pvalue_
BH=p.adjust(non_parametric_CD, method="BH"), stringsAsFactors=F)
stat_out_wilcox_df <- stat_out_wilcox_df[order(stat_out_wilcox_df$pvalue_BH),]

# How many species were significant in both tests?
sig_mgs_wilcox <- stat_out_wilcox_df$MGS[ stat_out_wilcox_df$pvalue_BH < 0.05 ]
sig_mgs_wilcox[ sig_mgs_wilcox %in% sig_mgs_lm ]
```

```
## [1] "MGS:373" "MGS:364" "MGS:511" "MGS:132" "MGS:183" "MGS:295" "MGS:126"
## [8] "MGS:59" "MGS:64" "MGS:5" "MGS:280" "MGS:9" "MGS:36" "MGS:58"
## [15] "MGS:25" "MGS:120"
```

Q: How many significant species overlap between the results generated using wilcox test and linear model? What is the difference between the two models?

Alternatively one can use a parametric model that is based on a **negative binomial distribution**. Implementation of that model is available in packages **DESeq2** or **edgeR**.

THE END